Assignment 4-1 Polynomial Addition 2

You are given two polynomials $f(x)$ and $g(x)$ with integer coefficients. In this problem you'll have to find out the sum $f(x) + g(x)$ of $f(x)$ and $g(x)$. For the sake of convenience, we define the zero polynomial as $0x^0$.

Input

The input consists of t ($30 \le t \le 40$) test cases. The first line of the input contains only positive integer t. Then t test cases follow. Each test case consists of two polynomials $f(x)$ and $g(x)$. Each polynomial consists of three lines: the first line contains only one positive integer m ($m \le 100$) which represents the number of nonzero terms of the polynomial; the second line consists of m nonzero integers (in the range $[-2^{62}, 2^{62} - 1]$) representing the coefficients of the polynomial; the third line consists of m integers (in the range $[0, 2^{62} - 1]$ and in decreasing order) representing the corresponding exponents of the polynomial. For example, the polynomial $300x^{1000} + 200x^{100} + 100x^{10}$ is represented as the following three lines:

```
3
300 200 100
1000 100 10
```

You may assume that neither $f(x)$ nor $g(x)$ is the zero polynomial.

Output

For each test case, you are to output three lines representing $f(x) + g(x)$, in the same format as the input. Note that if $f(x) + g(x)$ is the zero polynomial, the output should be as follows:

```
1
0
0
```

Sample Input

```
2
3
1 -2 -4
3 2 0
1
4
0
3
```

-1 -2 4
3 2 0
4
1 2 1 -7
3 2 1 0

Sample Output

2
1 -2
3 2
2
1 -3
1 0

Requirements

In your program, a polynomial $a_n x^{d_n} + a_{n-1} x^{d_{n-1}} + \cdots + a_2 x^{d_2} + a_1 x^{d_1} + a_0 x^{d_0}$ should be represented by the following two arrays:

	n	$n-1$	$\cdots$	2	1	0
coefficient array:	a_n	a_{n-1}	$\cdots$	a_2	a_1	a_0

	n	$n-1$	$\cdots$	2	1	0
exponent array:	d_n	d_{n-1}	$\cdots$	d_2	d_1	d_0

For example, the polynomial $-x^3 - 2x^2 + 4$ is represented by the following two arrays:

	2	1	0
coefficient array:	-1	-2	4

	2	1	0
exponent array:	3	2	0

Note that $d_n > d_{n-1} > \cdots > d_2 > d_1 > d_0 \geq 0$, and for every $i = 0, 1, \ldots, n$, $a_i \neq 0$.

Part of the program

You are required to write the function `addition` to complete the following program which solves this problem. In your program, you cannot declare global variables (including global arrays) except `arraySize`. Moreover, you cannot declare arrays (or vectors) in the function `addition`.

```
// Polynomial addition
#include <iostream>
using namespace std;

// sum = addend + adder
void addition( long long int addendCoef[], long long int addendExpon[], int
addendSize,
               long long int adderCoef[], long long int adderExpon[], int adderSize,
               long long int sumCoef[], long long int sumExpon[], int &sumSize );

// outputs the specified polynomial
void output( long long int coefficient[], long long int exponent[], int size );

const int arraySize = 100;

int main()
{
   int T;
   cin >> T;
   for( int t = 0; t < T; t++ )
   {
      long long int addendCoef[ arraySize ] = {};
      long long int addendExpon[ arraySize ] = {};
      int addendSize;
      cin >> addendSize; // input addend
      for( int i = addendSize - 1; i >= 0; i-- )
         cin >> addendCoef[ i ];
      for( int i = addendSize - 1; i >= 0; i-- )
         cin >> addendExpon[ i ];

      long long int adderCoef[ arraySize ] = {};
      long long int adderExpon[ arraySize ] = {};
      int adderSize;
      cin >> adderSize; // input adder
      for( int i = adderSize - 1; i >= 0; i-- )
         cin >> adderCoef[ i ];
      for( int i = adderSize - 1; i >= 0; i-- )
         cin >> adderExpon[ i ];

      long long int sumCoef[ 2 * arraySize ] = {};
      long long int sumExpon[ 2 * arraySize ] = {};
      int sumSize = 1;

      addition( addendCoef, addendExpon, addendSize,
                adderCoef, adderExpon, adderSize,
                sumCoef, sumExpon, sumSize );

      output( sumCoef, sumExpon, sumSize );
   }
}

// sum = addend + adder
void addition( long long int addendCoef[], long long int addendExpon[],
               int addendSize,
               long long int adderCoef[], long long int adderExpon[], int adderSize,
               long long int sumCoef[], long long int sumExpon[], int &sumSize )
{









}

// outputs the specified polynomial
void output( long long int coefficient[], long long int exponent[], int size )
{
   cout << size << endl;
   cout << coefficient[ size - 1 ];
   for( int i = size - 2; i >= 0; i-- )
      cout << " " << coefficient[ i ];
   cout << endl;

   cout << exponent[ size - 1 ];
   for( int i = size - 2; i >= 0; i-- )
      cout << " " << exponent[ i ];
```

```
    cout << endl;
}
```